第27卷 第1期　　工 程 数 学 学 报　　Vol. 27 No. 1
2010年02月　　CHINESE JOURNAL OF ENGINEERING MATHEMATICS　　Feb. 2010

文章编号:1005-3085(2010)01-0053-12

已知工件最大加工时间的三台同类机半在线问题*

罗润梓[1]，孙世杰[2]，何龙敏[2]

(1- 南昌大学理学院数学系，南昌 330031；2- 上海大学理学院数学系，上海 200444)

摘　要: 本文考虑已知工件最大加工时间的三台同类机半在线问题。三台机器的速度分别为 $s_1 = r$, $s_2 = 1$, $s_3 = s > 1$, $1 \leq r \leq s$，工件是一个一个独立地到来，工件的信息是逐个释放的，但所有工件中加工时间为最大的工件的加工时间是已知的，目标函数为极小化最大机器负载。本文证明任何解此问题的算法竞争比的下界为3/2且给出 $Q\max 3$ 算法并证明此算法的竞争比不大于 $\frac{2(r+s+1)}{2r+s}(1 < s \leq 2)$ 和 $\frac{r+2s+1}{r+s}(s > 2)$。

关键词: 排序；半在线；竞争比

分类号: **AMS**(2000) 90B35; 90C27　　**中图分类号**: O223　　**文献标识码**: A

1　引言

排序 (scheduling) 问题是研究在满足一定要求的条件下，如何安排有限资源以达到某种意义下的最优结果的研究分支，它是组合最优化领域中的一类重要问题，具有重要的理论意义和广泛实际背景的。它产生的主要背景是机器制造，后来被广泛应用于计算机系统、运输调度、生产管理等领域。根据我们对排序问题掌握的信息多少，可把排序问题分为三类：离线、在线、半在线。半在线排序问题是指排序者在排序前知道问题的部分信息，例如已知工件最大加工时间，已知工件总的加工时间等。这类既不是离线 (off-line)，又难以归为在线 (on-line) 的问题我们称之为半在线 (semi on-line) 问题。

自1996年半在线概念提出以后，已经出现了多种半在线模型[1,2]。在1999年，何和张[3]提出了一个已知工件最大加工时间的新半在线模型，这是一个研究得比较多的半在线模型。在本文中我们用 $*|$ known largest job $|*$ 来表示此模型，这里的第一个“$*$”表示机器环境，第二个“$*$”表示最优准则。文[3]给出了求解 $P2|$ known largest job $|C_{\max}$ 问题的最优算法PLS，其竞争比为4/3。文献[4]进一步证明了PLS算法也是求解 $P2|$ known largest job $|C_{\min}$ 问题的最优算法，其竞争比为3/2。对于 $Q2|$ known largest job $|C_{\max}$ 问题，文献[5]给出了竞争比为3/2的MLS算法，并证明此问题的下界为 $\sqrt{2}$。文献[6]进一步考虑了此问题，给出了 $Q\max 2$ 算法并证明此算法的参数竞争比分别为

$$\frac{2(s+1)}{s+2},\quad 1 \leq s \leq 2 \quad 和 \quad \frac{s+1}{s},\quad s > 2.$$

文献[7]则考虑了 $Q2|$ known largest job $|C_{\min}$ 问题，给出了 $C_{\min}$ 算法，并证明此算法的参数竞争比为 $\frac{2s+1}{s+2}$。当 $s = 1$ 时，其竞争比与文献[4]的结果是一致的。文献[8,9]分别考虑了三

收稿日期: 2008-01-28.　**作者简介**: 罗润梓 (1966年12月生)，男，博士，副教授. 研究方向：排序理论与应用、混沌控制与同步.

***基金项目**: 江西省自然科学基金 (2007GZS2126).

台机器的情形。文献[8]考虑了$Q3|$known largest job$|C_{\min}$问题，其中三台机器的速度分别为$s_1=s_2=1$，$1\le s=s_3$，给出了$C_{\min 3}$算法，并证明了此算法的竞争比为

$$\max\left\{2,\ \frac{3s+2}{2+s}\right\},$$

且当$1\le s\le 2$时算法为最优。对于更一般的情形，不妨假设三台机器的速度分别为$s_1=1, s_2=r, s_3=s, 1\le r\le s$，文[9]给出了min 3算法，并证明此算法的竞争比为

$$\max\left\{r+1,\ \frac{3s+r+1}{1+r+s}\right\}.$$

此问题的竞争比下界为$\max\{2,r\}$。由于在现实生活中精确地知道工件的最大加工时间是比较困难的，我们一般只能知道最大工件的加工时间在某一个范围比如属于$[p, rp]$内，这里r, p为大于0的常数。文献[10]考虑了这种情形，并称之为半在线模型的松弛，记为$*|$known largest jobinterval$|*$。文献[10]分别考虑了$P2|$known largest job interval$|C_{\max}$问题和$P2|$known largest jobinterval$|C_{\min}$问题，给出了PInterval算法并证明：

1) 当$1\le r\le 2$时，PInterval算法解$P2|$known largest job$|C_{\max}$问题的的竞争比为

$$\frac{2(1+r)}{2+r}.$$

当$r>2$时其竞争比为3/2且是最优的。

2) 当$1\le r\le 2$时，PInterval算法解$P2|$known largest job interval$|C_{\min}$问题的竞争为$1+\frac{r}{2}$；当$r>2$时其竞争比为2且是最优的。对于$P2|$known largest job interval$|C_{\max}$问题，文献[11]给出算法MPLS，并证明此算法的竞争比为

$$\frac{2(1+r)}{2+r},\quad 1\le r\le 2 \quad \text{且当} \quad r\in[1,\sqrt{5}-1]$$

时此算法为最优，同时给出了该问题的一个参数下界。

本文考虑$Q3|$known largest job$|C_{\max}$问题，这里三台机器M_1, M_2, M_3的速度分别假设为$s_1=r, s_2=1, s_3=s>1, 1\le r\le s$。本文的第二节给出$Q\max 3$算法，第三节证明$Q\max 3$算法的竞争比不大于

$$\frac{2(r+s+1)}{2r+s},\quad 1<s\le 2.$$

第四节证明$Q\max 3$算法解$Q3|$known largest job$|C_{\max}$问题的竞争比不大于

$$\frac{2(r+s+1)}{2r+s},\quad 1<s\le 2 \quad \text{和} \quad \frac{r+2s+1}{r+s},\quad s>2.$$

第五节是结论，证明了任何算法解此问题的竞争比的常数下界为3/2。

2 $Q\max 3$算法

在给出算法之前先定义几个符号，用$L(M_i)\,(i=1,2,3)$表示加工过程中某阶段机器M_i的负载，即当前已分给机器M_i加工的工件的总加工时间与机器速度的比值，在不致引起混淆的情况下我们仍用M_i表示。用$p_{\max}$表示最大的加工时间，某个工件在机器M_2上的加工时间

是 $p_{\max}$ 则称此工件是一个最大的工件仍用 $p_{\max}$ 表示，用 x 表示当前需要安排的工件及其所需加工时间。

下面给出 $Q\max 3$ 算法，$Q\max 3$ 算法是由NPLS算法和LS算法组合而成的。

$Q\max 3$ **算法**。

步骤1　如果 $1 < s \le 2$，则转步骤2。如果 $s > 2$，则转步骤3。

步骤2　根据NPLS算法加工所有工件。

步骤3　根据LS算法加工所有工件。

NPLS **算法**。

步骤1　如果

$$M_1 + \frac{x}{r} \le \frac{p_{\max}}{s},$$

则把当前工件 x 放在机器 M_1 上加工，否则转步骤2。

步骤2　如果

$$M_2 + x \le \frac{p_{\max}}{s},$$

则把 x 放在机器 M_2 上加工，否则转步骤3。

步骤3　如果

$$M_1 + \frac{x}{r} \ge \frac{2p_{\max}}{s}, \quad M_2 + x \ge \frac{2p_{\max}}{s},$$

则转步骤4。否则，若 $x \ne p_{\max}$，则在机器 M_1, M_2 上按LS算法安排工件 x；若 $x = p_{\max}$，如果

$$M_3 + \frac{x}{s} < \frac{2p_{\max}}{s},$$

则把 x 放在机器 M_3 上加工，如果

$$M_3 + \frac{x}{s} \ge \frac{2p_{\max}}{s},$$

则转步骤4。

步骤4　在机器 M_1, M_2, M_3 上按LS算法加工，即把工件 x 安排在能使其最快完工的机器上加工。

重复执行以上各步直到不再有新工件到来为止。在执行算法过程中，如果存在多台机器同时可以安排工件 x，则把它安排在从未安排过工件的机器上加工；若不存在这样的机器则把工件 x 安排在速度最快的机器上加工。

3　$Q\max 3$ 算法在 $1 < s \le 2$ 时的竞争比

本节假定 $1 < s \le 2$。由NPLS算法以及LS算法易知下面引理1、引理2成立。

引理1

1)　不论加工进行到哪一阶段都有

a)　$M_i - M_1 \le \frac{p_{\max}}{r}\ (i = 2, 3)$;　　b)　$M_i - M_2 \le p_{\max}\ (i = 1, 3)$ 成立；

2)　当 $M_3 \ge \frac{p_{\max}}{s}$ 时，不等式

$$M_i - M_3 \le \frac{p_{\max}}{s}, \quad i = 1, 2,$$

成立。

证明 首先证明结论 a)。

i) 若 $M_2 - M_1 > \frac{p_{\max}}{r}$，则 $M_2 - M_1 > \frac{p_{\max}}{s}$，从而此时安排在机器 M_2 上最后一个工件 (不妨设为 x) 是根据 LS 算法安排的，因此 $M_2 \le M_1 + \frac{x}{r}$，可得 $M_2 - M_1 \le \frac{p_{\max}}{r}$，与假设矛盾。

ii) 若 $M_3 - M_1 > \frac{p_{\max}}{r}$，则

$$M_3 > \frac{p_{\max}}{r} \ge \frac{p_{\max}}{s},$$

即 $M_3 > \frac{p_{\max}}{s}$，从而机器 M_3 上至少安排了两个工件。不妨设安排在机器 M_3 上的第一个工件为 x_{31}，最后一个工件为 x_{3f}。若 $x_{31} = p_{\max}$，此时 x_{31} 安排在机器 M_3 上是执行 NPLS 算法步骤 3 或步骤 4 的结果，不管是执行哪一步，x_{3f} 安排在机器 M_3 上一定是执行 NPLS 算法步骤 4 (即 LS 算法) 的结果，从而

$$M_1 + \frac{x_{3f}}{r} \ge M_3,$$

即

$$M_3 - M_1 \le \frac{x_{3f}}{r} \le \frac{p_{\max}}{r},$$

与假设矛盾。若 $x_{31} \ne p_{\max}$，如果 x_{3f} 是按 NPLS 算法步骤 3 安排的，则

$$x_{3f} = p_{\max}, \quad M_3 < \frac{2p_{\max}}{s}.$$

而由

$$M_1 + \frac{x_{31}}{r} \ge \frac{2p_{\max}}{s},$$

得

$$M_1 \ge \frac{2p_{\max}}{s} - \frac{x_{31}}{r},$$

从而

$$M_3 - M_1 < \frac{2p_{\max}}{s} - \left(\frac{2p_{\max}}{s} - \frac{x_{31}}{r}\right) < \frac{p_{\max}}{r},$$

与假设矛盾。如果 x_{3f} 是按 NPLS 算法步骤 4 安排的，则由 $M_1 + \frac{x_{3f}}{r} \ge M_3$，得

$$M_3 - M_1 \le \frac{p_{\max}}{r},$$

与假设矛盾。

其次证明结论 b)。设工件 x 是安排在机器 M_1 上的最后一个工件，若 x 是由步骤 1 安排的，则显然有 $M_1 \le \frac{p_{\max}}{r}$，因此 $M_1 - M_2 \le p_{\max}$。若 x 是由步骤 3 或步骤 4 安排的，由于此时对机器 M_1, M_2 来说都是由 LS 算法安排工件，因此有 $M_2 + x \ge M_1$，即 $M_1 - M_2 \le x \le p_{\max}$。

设工件 y 是安排在机器 M_3 上的最后一个工件，若 y 也是第一个工件，则当 $y = p_{\max}$ 时易知 $M_3 = \frac{p_{\max}}{s}$，显然

$$M_3 - M_2 \le \frac{p_{\max}}{s} \le p_{\max},$$

而当 $y \ne p_{\max}$ 时，则 y 是由步骤 4 安排的结果，此时 $y + M_2 \ge M_3$，可得 $M_3 - M_2 \le y \le p_{\max}$。若 y 不是第一个工件，则机器上至少安排了两个工件，下面类似于 a) ii) 可证得结论成立，此处略。

最后证明结论 2)。设工件 x 是安排在机器 M_1 上的最后一个工件，若 x 安排在机器 M_1 上是步骤 1 的结果，则 $M_1 \le \frac{p_{\max}}{s}$，显然

$$M_1 - M_3 \le \frac{p_{\max}}{s}.$$

若 x 安排在机器 M_1 上是步骤3的结果，则 $M_1 < \frac{2p_{\max}}{s}$，显然

$$M_1 - M_3 < \frac{p_{\max}}{s}.$$

若 x 安排在机器 M_1 上是步骤4的结果，则 $M_1 \le M_3 + \frac{p_{\max}}{s}$，可得 $M_1 - M_3 \le \frac{p_{\max}}{s}$。类似可证 $M_2 - M_3 \le \frac{p_{\max}}{s}$。

引理2　若 x 是按NPLS算法步骤4安排在机器 M_3 上加工的工件，则

$$x + rM_1 \ge \frac{2rp_{\max}}{s}, \quad x + M_2 \ge \frac{2p_{\max}}{s}.$$

引理3　设最后一个工件 p_n 到来前各机器 $M_i\,(i=1,2,3)$ 的负载分别为 M_1, M_2, M_3。

1)　如果 $M_3 \ge \frac{p_{\max}}{s}$，则

$$\frac{(r+s+1)\max\{M_1, M_2\}}{rM_1 + M_2 + sM_3 + p_n} < \frac{2(r+s+1)}{2r+s};$$

2)　如果 $M_3 > \frac{p_{\max}}{s}$，则

$$\frac{(r+s+1)M_3}{rM_1 + M_2 + sM_3 + p_n} < \frac{2(r+s+1)}{2r+s}.$$

证明　1)　根据引理1和引理3的已知条件，可得

$$s(M_1 - M_3) \le s \times \frac{p_{\max}}{s} = p_{\max} \le sM_3.$$

因此 $(2r+s)M_1 < 2(rM_1 + M_2 + sM_3 + p_n)$，从而

$$\frac{(r+s+1)M_1}{rM_1 + M_2 + sM_3 + p_n} < \frac{2(r+s+1)}{2r+s}.$$

又因为

$$2r(M_2 - M_1) \le 2r \times \frac{p_{\max}}{r} = 2p_{\max} \le 2sM_3,$$

故 $(2r+s)M_2 < 2(rM_1 + M_2 + sM_3 + p_n)$，从而

$$\frac{(r+s+1)M_2}{rM_1 + M_2 + sM_3 + p_n} < \frac{2(r+s+1)}{2r+s}.$$

2)　易知此时在机器 M_3 上至少有两个工件。不失一般性，设安排在机器 M_3 上的头两个工件按其加工时间大小排列为 x, $y\,(x \le y \le p_{\max})$。下面分两种情形来证明。

情形1　$\frac{p_{\max}}{s} < M_3 < \frac{2p_{\max}}{s}$.

显然 $x < p_{\max}$，且工件 x 安排在机器 M_3 上加工是执行算法步骤4的结果，因此由引理2知

$$x + rM_1 \ge \frac{2rp_{\max}}{s}, \quad x + M_2 \ge \frac{2p_{\max}}{s}.$$

由 $y \ge x$ 进一步可得

$$y + rM_1 \ge \frac{2rp_{\max}}{s}, \quad y + M_2 \ge \frac{2p_{\max}}{s}.$$

因为工件 x 和 y 都是安排在机器 M_3 上加工，故 $sM_3 \ge x + y$，从而

$$rM_1 + M_2 + sM_3 \ge rM_1 + M_2 + x + y \ge \frac{2rp_{\max}}{s} + \frac{2p_{\max}}{s}.$$

易知

$$(2r+s)M_3 < (2r+s)\times\frac{2p_{\max}}{s} \le \frac{2p_{\max}(2r+2)}{s},$$

因此 $(2r+s)M_3 < 2(rM_1+M_2+sM_3+p_n)$，故

$$\frac{(r+s+1)M_3}{sM_1+M_2+sM_3+p_n} < \frac{2(r+s+1)}{2r+s}.$$

情形 2 $M_3 \ge \frac{2p_{\max}}{s}$.

根据引理 1 可得

$$2r(M_3-M_1) \le 2r\times\frac{p_{\max}}{r} = 2p_{\max} \le sM_3,$$

从而 $(2r+s)M_3 < 2(rM_1+M_2+sM_3+p_n)$，因此

$$\frac{(r+s+1)M_3}{rM_1+M_2+sM_3+p_n} < \frac{2(r+s+1)}{2r+s}.$$

定理 1 $Q\max 3$ 算法解 $Q3|\text{known largest job}|C_{\max}$ $(1<s\le 2)$ 问题的竞争比不大于

$$\frac{2(r+s+1)}{2r+s}.$$

证明 当 $1<s\le 2$ 时，由于执行 $Q\max 3$ 算法其实是执行 NPLS 算法，因此我们把 $C_{Q\max 3}(J)$ 表示成 $C_{\text{NPLS}}(J)$。如果没有特别说明在本节以下部分中所说的算法均指 NPLS 算法。为方便起见，不妨设 p_n 是最后一个到达的工件，三台机器在 p_n 被安排之前的负载分别用 M_1, M_2, M_3 来表示。我们分三种情形考虑。

情形 1 $M_3 < \frac{p_{\max}}{s}$.

此时 $p_n = p_{\max}$，工件 p_n 应该安排在机器 M_3 上加工。且易知

$$M_1 < \frac{2p_{\max}}{s}, \quad M_2 < \frac{2p_{\max}}{s}, \quad C_{\text{NPLS}}(J) = \max\left\{M_1, M_2, M_3+\frac{p_{\max}}{s}\right\},$$

$$C^\star(J) \ge \frac{rM_1+M_2+sM_3+p_n}{1+r+s}.$$

1) 如果 $C_{\text{NPLS}}(J) = M_1$，则由

$$sM_1 - sM_2 = s(M_1-M_2) \le sp_{\max} \le 2s\Big(M_3+\frac{p_{\max}}{s}\Big),$$

可得

$$2rM_1+sM_1 \le 2rM_1+2M_2+2s\Big(M_3+\frac{p_{\max}}{s}\Big),$$

从而

$$\frac{C_{\text{NPLS}}(J)}{C^\star(J)} \le \frac{(1+r+s)M_1}{rM_1+M_2+sM_3+p_n} \le \frac{2(r+s+1)}{2r+s}.$$

2) 如果 $C_{\text{NPLS}}(J) = M_2$，则由

$$2r(M_2-M_1) \le 2r\times\frac{p_{\max}}{r} = 2p_{\max} \le 2s\Big(M_3+\frac{p_{\max}}{s}\Big),$$

可得

$$2rM_2+sM_2 \le 2rM_1+2M_2+2s\Big(M_3+\frac{p_{\max}}{s}\Big),$$

从而

$$\frac{C_{\text{NPLS}}(J)}{C^{\star}(J)} \le \frac{(1+r+s)M_2}{rM_1+M_2+sM_3+p_n} \le \frac{2(r+s+1)}{2r+s}.$$

3) 如果 $C_{\text{NPLS}}(J) = M_3 + \frac{p_{\max}}{s}$.

若 $M_3 = 0$，则 $M_1 \le \frac{p_{\max}}{s}$，$M_2 \le \frac{p_{\max}}{s}$，从而算法最优。

若 $M_3 > 0$，则 $M_2 > 0$，从而由引理2知 $sM_3 + M_2 \ge \frac{2p_{\max}}{s} \ge p_{\max}$。因为

$$2r\Big(M_3 + \frac{p_{\max}}{s} - M_1\Big) \le 2r \times \frac{p_{\max}}{r} = 2p_{\max} \le M_2 + sM_3 + p_{\max},$$

所以

$$(2r+s)\Big(M_3 + \frac{p_{\max}}{s}\Big) < 2rM_1 + 2M_2 + 2s\Big(M_3 + \frac{p_{\max}}{s}\Big),$$

从而

$$\frac{C_{\text{NPLS}}(J)}{C^{\star}(J)} \le \frac{(1+r+s)(M_3+\frac{p_{\max}}{s})}{rM_1+M_2+sM_3+p_n} < \frac{2(r+s+1)}{2r+s}.$$

注意到

$$\frac{C_{\text{NPLS}}(J)}{C^{\star}(J)} \le \frac{2(r+s+1)}{2r+s},$$

且当 $r=1$ 时 $\frac{2(r+s+1)}{2r+s} = 2$。但我们可以证明即使是 $r=1$，NPLS算法的竞争比还是小于2。这是因为此时 M_1，M_2，$M_3 + \frac{p_{\max}}{s}$ 都小于 $\frac{2p_{\max}}{s}$，而 $C^{\star}(J) \ge \frac{p_{\max}}{s}$，从而 $\frac{C_{\text{NPLS}}(J)}{C^{\star}(J)} < 2$。

情形2 $M_3 = \dfrac{p_{\max}}{s}$.

易知

$$M_i < \frac{2p_{\max}}{s}, \quad i = 1, 2.$$

如果工件 p_n 不被安排在机器 M_3 上，此时相当于情形1中 $M_3 = 0$ 的情形，因此仅需证明 p_n 安排在机器 M_3 上加工时结论成立。此时不管工件 p_n 是不是最大工件都是执行算法步骤4的结果，因此

$$M_1 + \frac{p_n}{r} \ge M_3 + \frac{p_n}{s}, \quad M_2 + p_n \ge M_3 + \frac{p_n}{s},$$

故

$$\begin{aligned}(2r+s)\Big(M_3+\frac{p_n}{s}\Big) &\le 2r\Big(M_1+\frac{p_n}{r}\Big) + s(M_2+p_n) \le 2(rM_1+p_n) + 2(M_2+p_n) \\ &= 2(rM_1+M_2+p_n+p_n) \le 2(rM_1+M_2+sM_3+p_n).\end{aligned}$$

从而

$$\frac{(r+s+1)(M_3+\frac{p_n}{s})}{rM_1+M_2+sM_3+p_n} \le \frac{2(r+s+1)}{2r+s}.$$

结合引理3可得

$$\frac{C_{\text{NPLS}}(J)}{C^{*}(J)} \le \frac{(r+s+1)\max\{M_1, M_2, M_3+\frac{p_n}{s}\}}{rM_1+M_2+sM_3+p_n} \le \frac{2(r+s+1)}{2r+s}.$$

注意到

$$\frac{C_{\text{NPLS}}(J)}{C^{\star}(J)} \le \frac{2(r+s+1)}{2r+s},$$

且当$r = 1$时$\frac{2(r+s+1)}{2r+s} = 2$。下面证明当$r = 1$时$\frac{C_{\mathrm{NPLS}}(J)}{C^*(J)} < 2$，只需证明当$p_n$安排在机器$M_3$上加工时结论成立即可。设$x$(其加工时间也用$x$表示)是已安排在机器$M_3$上且加工时间为最小的工件。

若$x < \frac{p_{\max}}{s}$，则由NPLS算法知

$$M_1 > \frac{p_{\max}}{s}, \quad M_2 > \frac{p_{\max}}{s}.$$

由

$$\frac{p_{\max}}{s} < M_3 + \frac{p_n}{s} \le \frac{2p_{\max}}{s},$$

知

$$C^*(J) > \frac{p_{\max}}{s}, \quad C_{\mathrm{NPLS}}(J) \le \frac{2p_{\max}}{s},$$

从而$\frac{C_{\mathrm{NPLS}}(J)}{C^*(J)} < 2$。

若$x = \frac{p_{\max}}{s}$，由于$\frac{x}{s} < \frac{p_{\max}}{s}$，故机器$M_3$上至少已经安排两个工件。又因为$\frac{3p_{\max}}{s^2} > \frac{p_{\max}}{s}$，因此机器$M_3$上有且只有两个工件。又易知$p_n = p_{\max}$，$M_1 > 0$，$M_2 > 0$。从而$C^*(J) > \frac{p_{\max}}{s}$，故$\frac{C_{\mathrm{NPLS}}(J)}{C^*(J)} < 2$。

若$x > \frac{p_{\max}}{s}$，此时易知机器M_3上有且只有一个工件，因而是最大工件。这是因为若机器M_3上已经安排有两个以上的工件，则$M_3 > \frac{2p_{\max}}{s^2} \ge \frac{p_{\max}}{s}$，这与$M_3 = \frac{p_{\max}}{s}$矛盾。

如果$p_n \le \frac{p_{\max}}{s}$，则

$$M_1 \ge \frac{p_{\max}}{s}, \quad M_2 \ge \frac{p_{\max}}{s}.$$

从而$C^*(J) > \frac{p_{\max}}{s}$，故$\frac{C_{\mathrm{NPLS}}(J)}{C^*(J)} < 2$。

如果$p_n > \frac{p_{\max}}{s}$，易知此时$M_1 > 0$，$M_2 > 0$。由

$$M_1 + \frac{M_2}{r} > \frac{p_{\max}}{s},$$

(否则，由NPLS算法知已安排在机器M_2上加工的工件都应该安排在机器M_1上加工，) 可得

$$C^*(J) > \frac{p_{\max}}{s}, \quad \frac{C_{\mathrm{NPLS}}(J)}{C^*(J)} < 2.$$

情形3 $M_3 > \frac{p_{\max}}{s}$.

1) p_n被安排在机器M_1上加工。由$M_1 + \frac{p_n}{r} \le M_2 + p_n$及$s \le 2$得

$$(2r+s)\left(M_1 + \frac{p_n}{r}\right) \le 2r\left(M_1 + \frac{p_n}{r}\right) + 2(M_2 + p_n) < 2(rM_1 + M_2 + sM_3 + p_n),$$

结合引理3可得

$$\frac{C_{\mathrm{NPLS}}(J)}{C^*(J)} \le \frac{(r+s+1)\max\{M_1 + \frac{p_n}{r}, M_2, M_3\}}{rM_1 + M_2 + sM_3 + p_n} < \frac{2(r+s+1)}{2r+s}.$$

2) p_n被安排在机器M_2上加工。

由$M_2 + p_n \le M_1 + \frac{p_n}{r}$及$s \le 2$得

$$(2r+s)(M_2 + p_n) \le 2r\left(M_1 + \frac{p_n}{r}\right) + 2(M_2 + p_n) < 2(rM_1 + M_2 + sM_3 + p_n),$$

结合引理3可得

$$\frac{C_{\mathrm{NPLS}}(J)}{C^*(J)}\le\frac{(r+s+1)\max\{M_1,M_2+p_n,M_3\}}{rM_1+M_2+sM_3+p_n}<\frac{2(r+s+1)}{2r+s}.$$

3)　p_n 被安排在机器 M_3 上加工。由

$$M_3+\frac{p_n}{s}\le M_1+\frac{p_n}{r},\quad M_3+\frac{p_n}{s}\le M_2+p_n,$$

及 $s\le 2$ 得

$$(2r+s)\Big(M_3+\frac{p_n}{s}\Big)\le 2r\Big(M_1+\frac{p_n}{r}\Big)+2(M_2+p_n)<2(rM_1+M_2+sM_3+p_n),$$

结合引理3可得

$$\frac{C_{\mathrm{NPLS}}(J)}{C^*(J)}\le\frac{(r+s+1)\max\{M_1,M_2,M_3+\frac{p_n}{s}\}}{rM_1+M_2+sM_3+p_n}<\frac{2(r+s+1)}{2r+s}.$$

至此，定理1获证。

4　$Q\max 3$ 算法在 $s>2$ 时的竞争比

定理2　$Q\max 3$ 算法解 $Q3|$ known largest job $|C_{\max}$ 问题的竞争比不大于

$$\frac{2(r+s+1)}{2r+s},\quad 1<s\le 2,\quad 和\quad \frac{r+2s+1}{r+s},\quad s>2.$$

证明　当 $1<s\le 2$ 时结论已经成立。当 $s>2$ 时，由于执行 $Q\max 3$ 算法其实是执行LS算法，因此我们把 $C_{Q\max 3}(J)$ 表示成 $C_{\mathrm{LS}}(J)$。不妨假设 $Q\max 3$ 算法结束时机器 M_i 的负载分别是 $M_i\,(i=1,2,3)$，下面分三种情形来讨论。

情形1　如果机器 M_2 的负载是最大的。

不妨设工件 y 是安排在机器 M_2 上的最后一个工件，$M_i^y\,(i=1,2,3)$ 表示工件 y 未安排前机器 M_i 的负载。显然

$$M_1^y+\frac{y}{r}\ge M_2,\quad M_3^y+\frac{y}{s}\ge M_2.$$

又因为 $M_1\ge M_1^y$，$M_3\ge M_3^y$，故 $M_1\ge M_2-\frac{y}{r}$，$M_3\ge M_2-\frac{y}{s}$。结合 $M_2\ge y$ 便有

$$\frac{(r+s+1)M_2}{rM_1+M_2+sM_3}\le\frac{(r+s+1)M_2}{(r+s+1)M_2-2y}\le\frac{(r+s+1)y}{(r+s+1)y-2y}=\frac{r+s+1}{r+s-1}<\frac{r+2s+1}{r+s}.$$

因此

$$\frac{C_{\mathrm{LS}}(J)}{C^*(J)}\le\frac{(r+s+1)\max\{M_1,M_2,M_3\}}{rM_1+M_2+sM_3}=\frac{(r+s+1)M_2}{rM_1+M_2+sM_3}<\frac{r+2s+1}{r+s}.$$

情形2　如果机器 M_1 的负载是最大的。

1)　若

$$M_1<\frac{(r+2s+1)p_{\max}}{(r+s)s},$$

显然

$$R_{\mathrm{LS}}<\frac{r+2s+1}{r+s}.$$

2) 若

$$M_1 \geq \frac{(r+2s+1)p_{\max}}{(r+s)s},$$

此时易知 $M_3 \geq M_1 - \frac{p_{\max}}{s}$。因此

$$\frac{(r+s+1)M_1}{rM_1+M_2+sM_3} \leq \frac{(r+s+1)M_1}{rM_1+s(M_1-\frac{p_{\max}}{s})} = \frac{(r+s+1)M_1}{(r+s)M_1-p_{\max}}$$

$$\leq \frac{(r+s+1)\frac{(r+2s+1)p_{\max}}{(r+s)s}}{(r+s)\frac{(r+2s+1)p_{\max}}{(r+s)s}-p_{\max}} = \frac{r+2s+1}{r+s}.$$

故

$$\frac{C_{\mathrm{LS}}(J)}{C^*(J)} \leq \frac{(r+s+1)\max\{M_1,M_2,M_3\}}{rM_1+M_2+sM_3} = \frac{(r+s+1)M_1}{rM_1+M_2+sM_3} \leq \frac{r+2s+1}{r+s}.$$

注意到当 $r=1$ 时 $\frac{r+2s+1}{r+s}=2$，下面证明此时 $\frac{C_{\mathrm{LS}}(J)}{C^*(J)}<2$。结论是显然成立的，因为当 $r=1$ 时机器 M_1 与机器 M_2 的作用是等价的。由情形1知道此时

$$\frac{C_{\mathrm{LS}}(J)}{C^*(J)} < \frac{r+s+1}{r+s-1} < 2.$$

情形3 如果机器 M_3 的负载是最大的，与情形2类似可得

$$\frac{C_{\mathrm{LS}}(J)}{C^*(J)} \leq \frac{(r+s+1)\max\{M_1,M_2,M_3\}}{rM_1+M_2+sM_3} = \frac{(r+s+1)M_3}{rM_1+M_2+sM_3} \leq \frac{r+2s+1}{r+s}.$$

下面证明当 $r=1$ 时 $\frac{C_{\mathrm{LS}}(J)}{C^*(J)}<2$。显然下不等式成立

$$\frac{C_{\mathrm{LS}}(J)}{C^*(J)} \leq \frac{(s+2)\max\{M_1,M_2,M_3\}}{M_1+M_2+sM_3} = \frac{(s+2)M_3}{M_1+M_2+sM_3} \leq \frac{2+s}{s} < 2.$$

结合定理1，知定理2成立。

5 结论

由文献[12]知当 $s_i=1\,(i=1,2)$，$s_3=s>1$ 时，LS算法是解 $Q3||C_{\max}$ 问题的最优算法，其竞争比为2。比较自然地会想到LS算法解 $Q3|\,\text{known largest job}\,|C_{\max}$ 问题的竞争比会不会有所改进？下面的实例说明LS算法的竞争比仍为2。不妨假设机器的速度分别为 $s_1=1+\varepsilon$，$s_2=1$，$s_3=s=2-\varepsilon$（ε 为很小的正数），最大工件的加工时间为 $\frac{2(1+2\varepsilon)}{(1-2\varepsilon)^2}$。考虑工件集

$$\left\{p_1=1,\, p_2=\frac{1+2\varepsilon}{1-2\varepsilon},\, p_3=\frac{2(1+2\varepsilon)}{(1-2\varepsilon)^2}\right\}.$$

显然按LS算法此三个工件都应该放在机器 M_3 上加工。此时

$$C_{\mathrm{LS}} = \frac{1+\frac{1+2\varepsilon}{1-2\varepsilon}+\frac{2(1+2\varepsilon)}{(1-2\varepsilon)^2}}{2-\varepsilon}, \quad C^* = \frac{2(1+2\varepsilon)}{(1-2\varepsilon)^2(2-\varepsilon)},$$

从而 $R_{\mathrm{LS}}=2(\varepsilon\to 0)$。

上例表明即使是已知工件的最大加工时间，LS算法的竞争比也不会改进。如果此例是按$Q\max 3$算法安排工件则$Q\max 3$算法是最优算法，由此可知$Q\max 3$算法与LS算法相比具有一定的优越性。这是因为$Q\max 3$算法是在速度最快的机器上预先留有一段空间$(\frac{2p_{\max}}{s})$放最大工件，但预留空间的时间不会太久。如果最大工件迟迟不到而当前工件安排在前两台机器上又会使得它们的负载大于或等于$\frac{2p_{\max}}{s}$，此时不再等待而是按LS算法把当前工件安排在能够使其最早完工的机器上加工。这样就使得各台机器的负载相差不会太大，避免了当安排完最后到来的最大工件时机器的负载相差较大这种情形。因此若最大工件来得较晚则$Q\max 3$算法优越性更明显。上例中如果工件集为

$$\left\{p_1 = \frac{2(1+2\varepsilon)}{(1-2\varepsilon)^2}, p_2 = \frac{1+2\varepsilon}{1-2\varepsilon}, p_3 = 1\right\},$$

则两算法都是最优的。$Q\max 3$算法的竞争比的优越性还表现在其是参数形式且含有两个变量，一般竞争比参数形式都是一个变量很少有含两个参数的竞争比。根据$Q\max 3$算法竞争比的表达形式我们可以求出不同的机器速度所对应的竞争比。为了更好地比较$Q\max 3$算法的竞争比我们给出问题的常数下界3/2。此下界可由下面的实例得到：假设机器的速度分别为$s_1 = 1+\varepsilon$，$s_2 = 1$，$s_3 = s = 2\,(\varepsilon > 0)$，最大工件的加工时间为2。考虑前三个工件$p_1 = 1$，$p_2 = 1$，$p_3 = 2$。若算法$A$把把其中两个工件安排在同一台机器上，则$C_A \geq \frac{3}{2}$，而$C^* = 1$，从而$R_A \geq \frac{3}{2}$。故不妨假设三个工件分别安排在不同的机器上，且易知最大工件应该安排在机器M_3上。此时最后两个最大工件到来，不论如何安排这两个工件，$C_A \geq \frac{3}{1+\varepsilon}$，而$C^* = 2$，从而$R_A \geq \frac{3}{2}$。

由于问题本身所具有的复杂性，我们仅给出了任何算法解$Q3|\,\text{known largest job}\,|C_{\max}$问题竞争比的常数下界为3/2而非参数形式的竞争比下界，因此给出一个较好的参数竞争比下界是一个值得进一步研究的问题。另外我们这里考虑的问题仅仅是当$m = 3$时的情形，是否可以把结果推广到更一般的$m \geq 4$情形呢？这也是一个值得深入研究的课题。

参考文献：

[1] 何勇，杨启帆，谈之奕. 平行机半在线排序问题研究(I)[J]. 高校应用数学学报，2003, 18A: 105-114
He Y, Yang Q F, Tan Z Y. Semi on-line scheduling on parallel machines(1)[J]. Appl Math J Chinese University, 2003, 18A: 105-114

[2] 何勇，杨启帆，谈之奕. 平行机半在线排序问题研究(II)[J]. 高校应用数学学报，2003, 18A: 213-222
He Y, Yang Q F, Tan Z Y. Semi on-line scheduling on parallel machines(II)[J]. Appl Math J Chinese University, 2003, 18A: 213-222

[3] He Y, Zhang G. Semi on-line scheduling on two identical machines[J]. Computing, 1999, 62: 179-187

[4] He Y. Semi on-line scheduling problem for maximizing the minimum machine completion time[J]. Acta Mathematica Applicate Sinica, 2001, 17: 107-113

[5] 谈之奕. 同类机半在线排序问题及相关问题研究[D]. 浙江大学博士学位论文，2001
Tan Z Y. The studies of online and semi online parallel machine scheduling and their related problems[D]. *Doctoral Dissertation of Zhejiang University*, 2001

[6] 罗润梓，孙世杰. 已知工件最大加工时间的两台同类机半在线问题[J]. 系统科学与数学，2006, 26(6): 729-736
Lou R Z, Sun S J. Semi on-line scheduling problem with known the largest processing time of jobs on two uniform machines[J]. J Sys Sci & Math Scis, 2006, 26(6): 729-736

[7] Luo R Z, Sun S J, Huang W P. Semi on-line scheduling problem for maximizing the minimum machine completion time on two uniform machines[J]. Journal of Systems Science and Complexity, 2006, 19: 101-107

[8] Luo R Z, Sun S J. Semi on-line scheduling problem for maximizing the minimum machine completion time on three special uniform machines[J]. Asia-Pacific Journal of Operational Research, 2005, 22: 229-237

[9] Luo R Z, Sun S J. Semi on-line scheduling problem for maximizing the minimum machine completion time on three uniform machines[J]. Journal of Zhejiang University, 2005, 6A(6): 591-595

[10] 罗润梓，孙世杰. 半在线模型的松弛[J]. 应用科学学报，2007, 25(5): 535-540
Luo R Z, Sun S J. Relaxation of semi-online model[J]. Journal of Applied Sciences, 2007, 25(5): 535-540

[11] Tan Z Y, He Y. *Semi-online scheduling problems on two identical machines with inexact partial information*[J]. Theoretical Computer Science, 2007, 377: 110-125

[12] Li R, Shi L. An on-line algorithm for some uniform processor scheduling[J]. SIAM J on Computing, 1998, 27: 414-422

Semi-online Scheduling Problem on Three Uniform Machines with the Known Largest Job

LUO Run-zi[1], SUN Shi-jie[2], HE Long-min[2]

(1- Department of Mathematics, Nanchang University, Nanchang 330031;
2- Department of Mathematics, Shanghai University, Shanghai 200444)

Abstract: In this paper, we investigate a semi-online scheduling problem on three uniform machines. Three speeds $s_1 = r$, $s_2 = 1$, $s_3 = s > 1$, $1 \leq r \leq s$ are associated with machine $M_i\,(i = 1, 2, 3)$, respectively. The jobs arrive one by one without knowledege of sucessive jobs, but the processing time of the largest job is known in advance. Our goal is to minimize the $C_{\max}$-the maximum workload of three machines. For this problem, we give a $Q\max 3$ algorithm and prove its competitive ratio is not greater than $\frac{2(r+s+1)}{2r+s}\,(1 < s \leq 2)$ and $\frac{r+2s+1}{r+s}\,(s > 2)$, while its lower bound is 3/2.

Keywords: scheuling; semi-online; competitive ratio

Received: 28 Jan 2008. **Accepted:** 16 July 2008.
Foundation item: The Natural Science Foundation of Jiangxi Province (2007GZS2126).